Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of claims:

1. (Withdrawn) A front-end unit for a processor, comprising:

an instruction cache system having an input for new addresses and an output for decoded instructions

an access filter coupled to the input for new addresses,

a segment builder having an input coupled to the instruction cache system's output, the segment builder to be disabled selectively by the access filter, and

a segment cache coupled to the segment builder.

2. (Withdrawn) The front-end unit of claim 1, wherein the instruction cache system comprises an instruction cache and an instruction decoding system coupled to the instruction cache.

3. (Withdrawn) The front-end unit of claim 2, wherein the instruction cache and the access filter are mutually integrated.

4. (Withdrawn) The front-end unit of claim 1, wherein the access filter comprises:

an address decoder,

a plurality of filter entries to store tag values, the filter entries coupled to the address decoder, and

a comparator coupled to an addressing input and to the filter entries.

5. (Original) A control method comprising, on a cache hit:

counting a number of accesses to a cache line that caused the hit,

if the count meets a predetermined threshold, enabling a segment builder,

building and storing instruction segments from an output of the segment builder.

6. (Original) The control method of claim 5, further comprising maintaining the segment builder in an unpowered state except in response to the cache hit.

7. (Original) The control method of claim 5, further comprising, if a hit also is registered in a segment cache, maintaining the segment builder disabled regardless of the count value.

8. (Original) The control method of claim 5, further comprising incrementing the count value and storing the incremented count value in the cache line.

9. (Original) The control method of claim 5, further comprising:

identifying a victim cache line, and

reducing a count value of the victim cache line.

10. (Original) The control method of claim 5, further comprising:

identifying an age of cache lines in a same set as the cache line that caused the hit, and

reducing a count value of those cache lines that are older than a median age of all the cache lines in the same set.

11. (Previously presented) An instruction cache, comprising:

an address decoder;

a plurality of cache entries, each indexed by an output of the address decoder and comprising a tag field, a count field and a data field;

an incrementor coupled to the access count fields; and

a threshold comparator coupled to the incrementor.

12. (Original) The cache of claim 11, further comprising:

a tag comparator coupled to the tag fields,

a transmission gate coupled to the threshold comparator and controlled by an output from the tag comparator.

13. (Original) The cache of claim 12, further comprising write control logic controlled by an output of the tag comparator.

14. (Original) The cache of claim 12, further comprising an eviction unit, wherein data in a count field is reduced when the eviction unit identifies the count field's cache entry as a victim.

15. (Original) The cache of claim 12, further comprising

a plurality of ways and set, and

an eviction unit having an age matrix, wherein data in a count field is reduced when the age matrix identifies data associated with the count field as being older than at least half the data stores in same set of other ways.

16. (Previously presented) An instruction cache, comprising:

an address decoder;

a plurality of cache entries, each indexed by an output of the address decoder and comprising a tag field, a count field and a data field;

a threshold comparator coupled to the count fields, and

a incrementor coupled to the threshold comparator.

17. (Original) The cache of claim 16, further comprising:

a tag comparator coupled to the tag fields,

a transmission gate coupled to the incrementor and controlled by an output from the tag comparator.

18. (Original) The cache of claim 16, further comprising an eviction unit, wherein data in a count field is reduced when the eviction unit identifies the count field's cache entry as a victim.

19. (Original) The cache of claim 16, further comprising

a plurality of ways and set, and

an eviction unit having an age matrix, wherein data in a count field is reduced when the age matrix identifies data associated with the count field as being older than at least half the data stores in same set of other ways.

20. (Original) The cache of claim 16, further comprising write control logic controlled by an output of the tag comparator.

21. (Previously presented) An access filter coupled to an instruction cache, the access filter comprising:

an address decoder;

a plurality of entries, each indexed by an output of the address decoder and comprising a tag field;

a tag comparator coupled to the tag fields,

wherein an output of the tag comparator enables an instruction segment builder.

22. (Original) The access filter of claim 21, further comprising:

a count field provided within each of the entries,

an incrementor coupled to the count fields, and

a threshold comparator coupled to the incrementor.

23. (Original) The access filter of claim 21, further comprising:

a count field provided within each of the entries,

a threshold comparator coupled to the count fields, and

an incrementor coupled to the incrementor.

24. (Original) The access filter of claim 21, further comprising a write controller coupled to an output of the incrementor.

25. (Canceled)

26. (New) An apparatus comprising:

a cache including a cache line having an access count field to store a count of a number of accesses to the cache line;

a comparator to determine whether the count meets or exceeds a predetermined threshold, and

an enable signal to enable a segment builder based on the count meeting or exceeding the predetermined threshold, the segment builder to build and store instruction segments.

27. (New) The apparatus of claim 26, further comprising an incrementor to increment the count.

28. (New) The apparatus of claim 26, the cache line further including a data field.

29. (New) The apparatus of claim 28, the cache line further including a tag field to store data representing an address of program instructions stored in the data field.

30. (New) The apparatus of claim 27, wherein the incrementor is a non-saturating incrementor.

31. (New) A system comprising;

an instruction cache;

an instruction segment builder; and

an access filter, the access filter including logic to

on a hit in the instruction cache, count a number of accesses to a cache line that caused the hit, and

if the count meets or exceeds a predetermined threshold, enable the segment builder to build and store instruction segments.

32. (New) The system of claim 31, further comprising a segment cache coupled to the segment builder.

33. (New) The system of claim 31, further comprising an instruction synchronizer coupled to the instruction cache.

34. (New) The system of claim 33, further comprising an instruction decoder coupled to the instruction synchronizer.